Slot-based Bandwidth Reservation in the Clipper Project[*]

Gary Hoo and William Johnston, Lawrence Berkeley National Laboratory

Ian Foster and Alain Roy, Argonne National Laboratory and University of Chicago

[*] This work is supported by the U. S. Dept. of Energy, Energy Research Division, Mathematical, Information, and Computational Sciences office, under contract DE-AC03-76SF00098 with the University of California and contract XXX with the University of Chicago. The authors may be contacted at: gjhoo@lbl.gov, wejohnston@lbl.gov, itf@mcs.anl.gov, roy@mcs.anl.gov. This document is report LBNL-xxxxx. For more information see http://www.itg.lbl.gov/Clipper/ .

The problem being addressed

Support for solving problems in *grid* (high-performance, distributed, internetworked) computing environments that require aggregating many resources.

One aspect of this is the network bandwidth needed to connect other resources (e.g. CPUs, storage systems, instrument systems, etc.) that must act in concert.

This requires network quality of service in shared networks that:

- **provides high-priority bandwidth (lossless, low delay)**
- **is reservable in advance**
- **is managed by a system that can participate in negotiation**
- **is not limited to low bandwidths**

The physical model

- **Sources and sinks (C1,2) at multiple sites**
- **Multiple network service providers (NSP1,2,3) some of which might be site LANs**
- **NSP ingress nodes (I1,2,...) that provide**
  - **traffic conditioning**
  - **policy-based access control**
  - **accounting**
- **Restriction points (R1,....) in the interiors of the networks that must be scheduled**

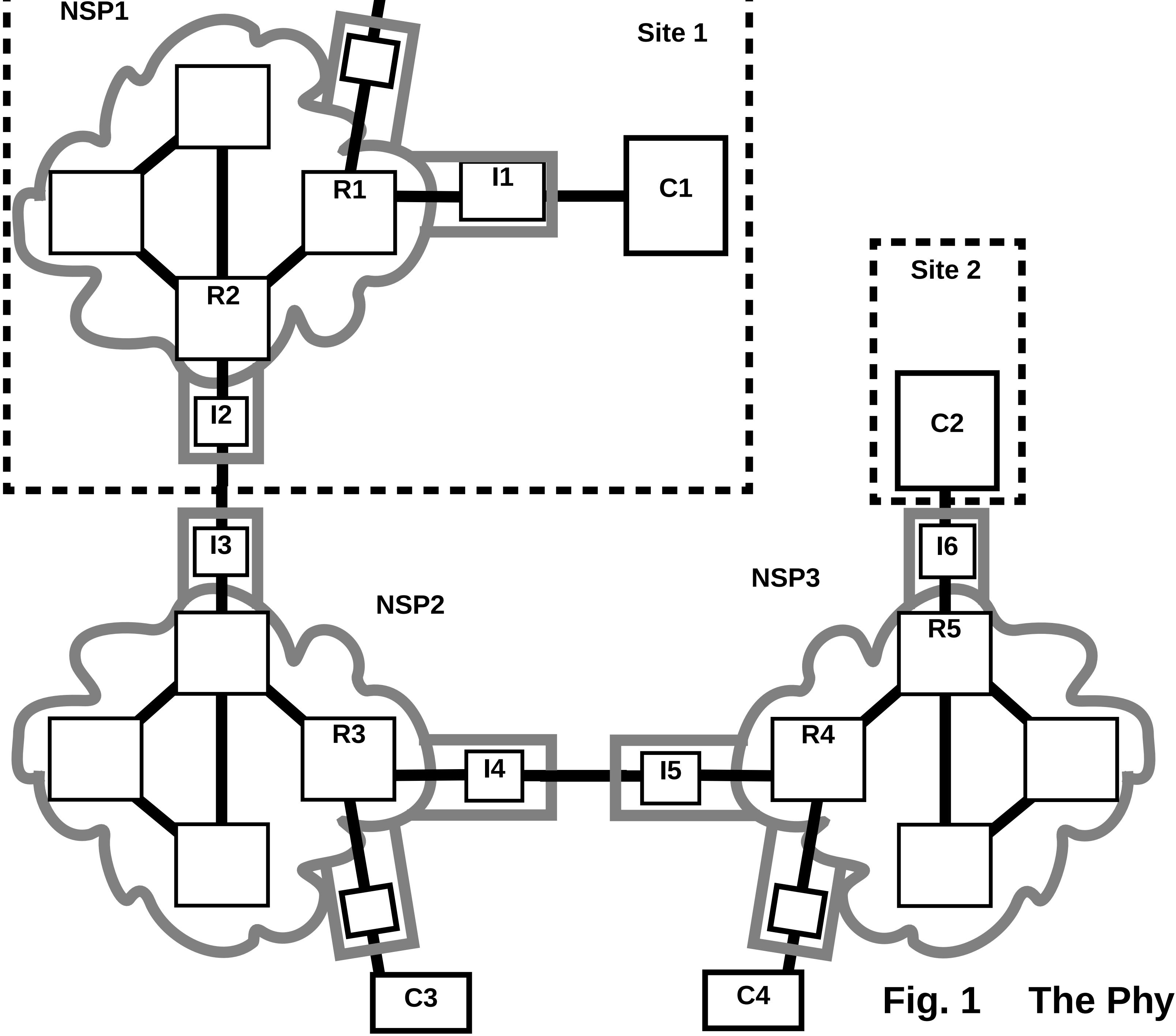


Fig. 1   The Physical Model

Approach to bandwidth reservation

Middleware services manage access to, and use of, diffserv priority service classes

- **Access control for service classes (admission control) provided by a reservation system separate from the router services**
- **The *resource* being reserved is a class of (priority) service**
- **Basic reservation unit is a *slot*:**
    - **some amount of bandwidth in some service class**
    - **a start and end time (whether end time is soft or hard is policy issue)**
    - **cf. Olov Schelén's work**
- ***Slot manager* keeps track of allocated slots and amount of bandwidth remaining to be allocated in the service class, effectively limiting amount of bandwidth that can be used in the class**

Approach to bandwidth reservation (cont.)

- Slot managers are in the trust domain of (are controlled by) the NSP

- Requests for slots are mediated by an access control system that implements the policy for accessing premium bandwidth

- Use of slots (claiming of reservation) is recorded in an accounting record

- Note: service-level agreements are static

  (That is, adjacent NSPs will have a “business” agreement that says, e.g., NSP1 can automatically use a service class in NSP2 that provides up to X bits/sec of high priority traffic (e.g. at I3 and I5). Therefore, use of that class is just a slot reservation issue, not a dynamic or SLA-level “policy” issue.)

Approach to bandwidth reservation (cont.)

Design goals

- Mechanism for advance, end-to-end reservation of bandwidth
- Mechanism for (simple) “negotiation”
  (I.e., a way to query for available priority bandwidth within some given range of bandwidth and time periods to allow a broker function to choose the best slot.)
- Mechanism for end-to-end reservation path discovery
- Preemptive reservation cancellation mechanism with notification
  (in case of unforeseen path change invalidating reservation)
- “Claiming” (of reservation, at flow start-up) must be lightweight

Approach to bandwidth reservation (cont.)

Design goals (cont.)

- **Network service provider has direct control over all resource utilization within its domain**
- **Policy-based access control, per flow, to premium service**

  Policy is determined by *stakeholders* (those who exercise administrative control over the premium service). This is in addition to policies for traffic aggregates expressed by SLA and will usually be finer-grained than SLA policies.
- **Accounting for use of priority service**

Approach to bandwidth reservation (cont.)

Elements of the architecture

- **Slot manager**
  - **implements resource behavior policy (e.g., no oversubscription of premium service class—oversubscription changes service's behavior)**
  - **performs basic reservation functions (allocating and de-allocating slots)**

Approach to bandwidth reservation (cont.)

Elements of the architecture (cont.)

- **Resource interface module**
  - **manages physical component that controls resources to be allocated (such as a single router or switch, or a collection of these that are scheduled as a unit because they represent a “restriction point” in the net)—e.g., causes state to be established in router to mark a flow**

Approach to bandwidth reservation (cont.)

Elements of the architecture (cont.)

- ♦ **Policy-based access control engine**
  - **Akenti policy engine provides fine-grained access control using digitally-signed documents (*certificates*)**
  - **Akenti allows stakeholders to define their policy in terms of *use-conditions***
  - **Users obtain *identity* (X.509) and *attribute certificates***
  - **Akenti collects use-condition, identity and attribute certificates for each access request to a protected resource; if use-conditions are satisfied by identity and attribute certificates, access is granted**

Approach to bandwidth reservation (cont.)

Elements of the architecture (cont.)

- **Resource manager (RM)**
  - **brokerage interface to slot manager**
  - **provides simple negotiation**
  - **may represent multiple slot managers and resource interface modules**
  - **understands network topology in order to provide identity of next RM that must be contacted to form a *reservation path***
  - **under control of the resource owner (NSP)**
  - **can invoke policy-based access control engine to determine whether to allocate a slot to a particular user**

Approach to bandwidth reservation (cont.)

Elements of the architecture (cont.)

- ♦ Resource manager (RM) (cont.)
  - invokes resource interface module at claim time to enable use of reserved resource
  - RM at ingress point emits secure accounting record when reservation is claimed

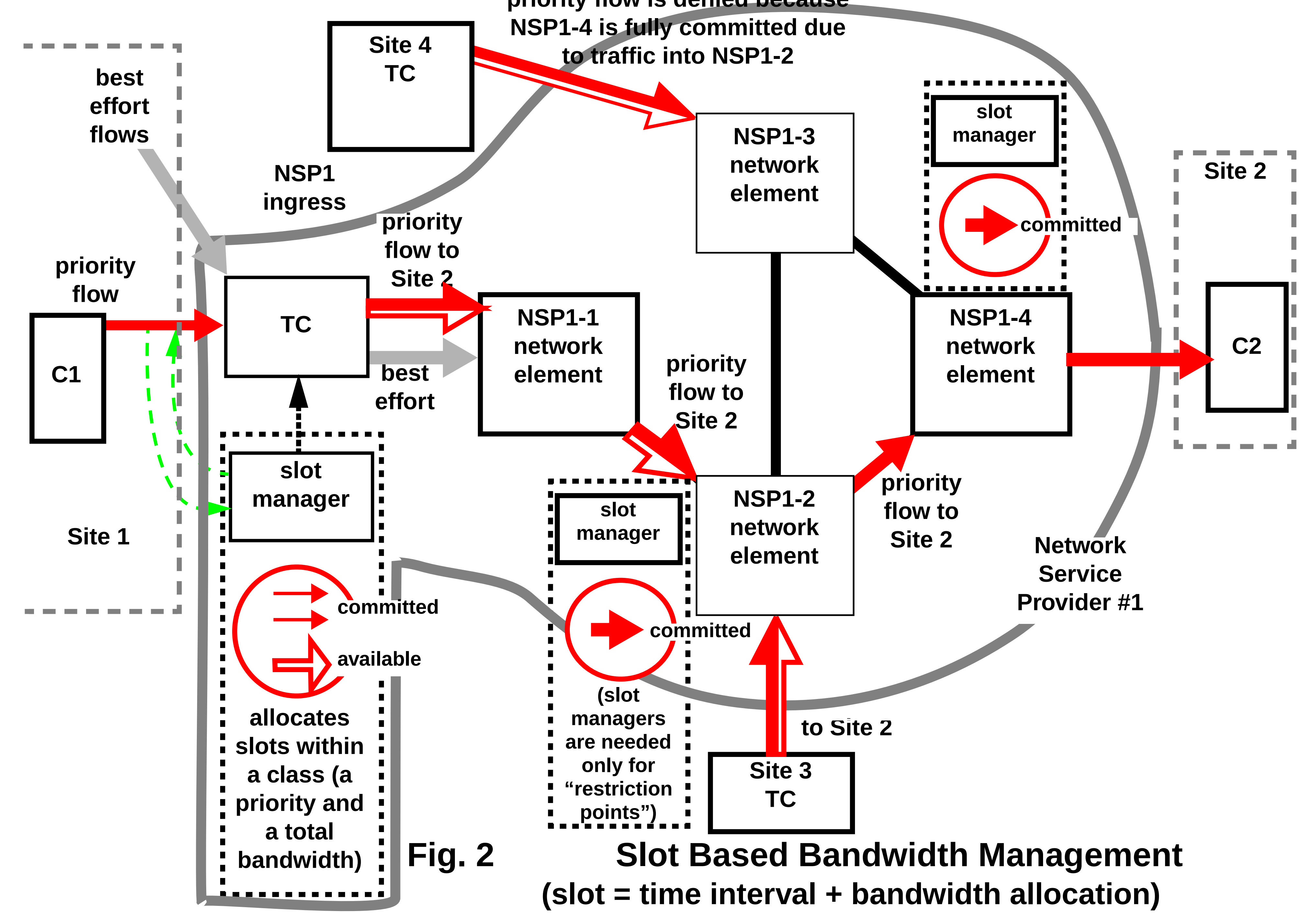


Fig. 2 **Slot Based Bandwidth Management**

(slot = time interval + bandwidth allocation)

Approach to bandwidth reservation (cont.)

Elements of the architecture (cont.)

- **Broker**
  - **general resource negotiator and aggregator**
  - **responsible for coordinating reservations on all of the resources (bandwidth on network paths, CPUs on multiple systems, etc.) required to accomplish a task**
  - **has to be able to query resources for available slots so that it may find a common time interval over all required resources**
  - **follows hop-by-hop path through network according to each RM’s “next contact” information**

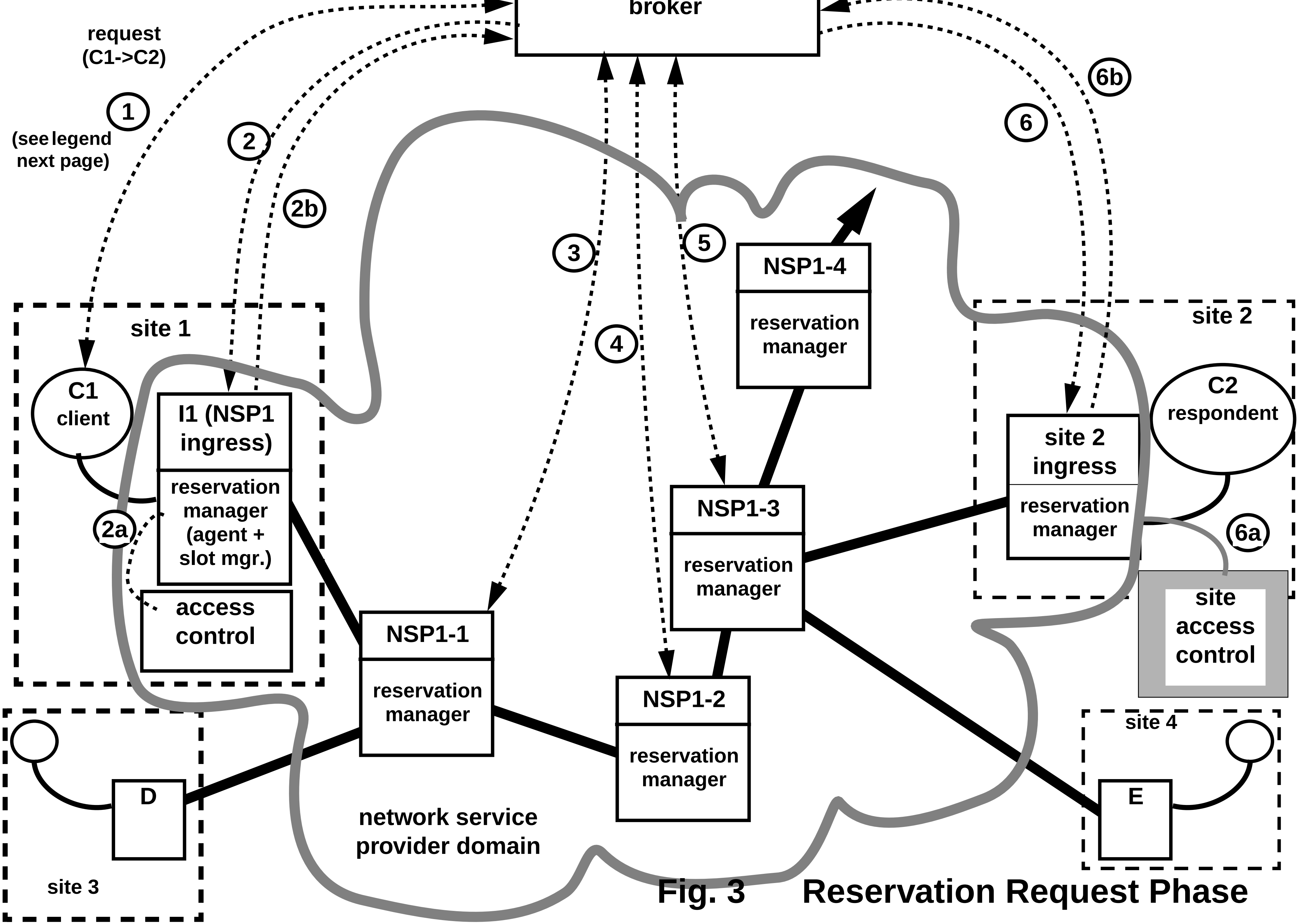


Fig. 3 Reservation Request Phase

1) client C1 asks broker for premium bandwidth to C2

2) broker makes request of NSP ingress router RM with *client_id* (e.g., user’s X.509 cert)

2a) ingress RM uses *client_id* to verify authority to use resource

2b) ingress RM returns to broker

- **reservation token**
  - ***client_id***
  - **authorization (as certified by ingress RM)**
  - **ingress resource reservation**
- ***next_rm* (next RM to contact)**

3) broker presents ingress reservation token to *next_rm* which:

- **validates token**
- **makes reservation against its allocation pool**
- **returns its own reservation token and *next_rm* to broker**

4) broker presents prior RM’s reservation token to *next_rm*, etc.

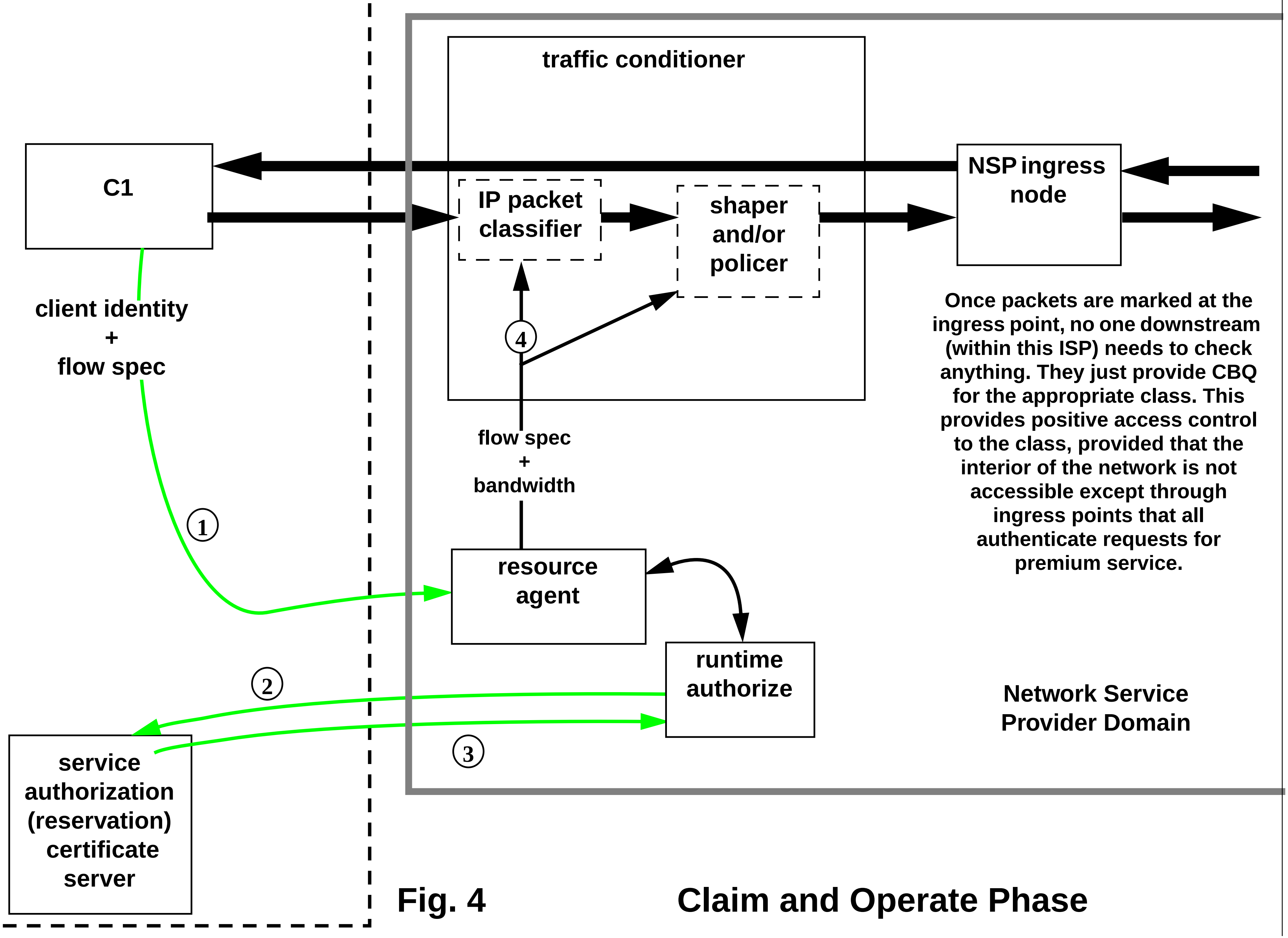
5) broker presents prior RM’s reservation token to *next_rm*, etc.

(any individual reservation failure invalidates overall reservation)

6) broker presents reservation token to *site_2* ingress RM

6a) *site_2* ingress RM requests authorization from *site_2* access control system to use this resource (based on proxy authorization previously given by C2 to C1 out of band)

6b) *site_2* ingress RM returns reservation token to broker with no *next_rm*



traffic conditioner

C1

IP packet classifier

shaper and/or policer

NSP ingress node

client identity + flow spec

1

4

flow spec + bandwidth

resource agent

runtime authorize

2

3

service authorization (reservation) certificate server

Network Service Provider Domain

Once packets are marked at the ingress point, no one downstream (within this ISP) needs to check anything. They just provide CBQ for the appropriate class. This provides positive access control to the class, provided that the interior of the network is not accessible except through ingress points that all authenticate requests for premium service.

Fig. 4 Claim and Operate Phase

Status

- ♦ Slot manager
  - prototype implemented (Alain Roy)
- ♦ Resource interface module
  - prototype implemented in 1997 for SC97 demo (Van Jacobson)
  - needs to be revamped
- ♦ Akenti
  - interface completed
  - Akenti in use in Diesel Combustion Collaboratory

Status (cont.)

- Resource manager
  - interface designed (for Globus)
  - required functionality and near-term importance () identified:
    - (1) ability to identify “next agent to contact” - that is, given the destination address what is the next restriction point along the path
    - (3) respond with available slots “close” to the request
    - (3) manage several restriction points (several slot managers)
    - (2) ability to invoke policy engine to authorize user
    - (1) returns a token representing the reservation

Status (cont.)

- **Broker**
    - **utility of Globus DUROC needs to be determined (i.e., can DUROC determine a common slot for multiple resources)**
    - **RSL extended for bandwidth slots (Foster, et al.)**